

An Efficient General-Purpose Least-Squares Refinement Program for Macromolecular Structures

Dale E. Tronrud,
Lynn F. Ten Eyck*, and
Brian W. Matthews

Institute of Molecular Biology
and
Departments of Chemistry and Physics
University of Oregon
Eugene, OR 97403, USA

Received: 16 July 1986
Accepted: 12 November 1986

*Present address: 79521 Highway 99N, Cottage Grove, Oregon 97424, USA

Abstract

A package of programs has been developed for efficient restrained least-squares refinement of macromolecular crystal structures. The package has been designed to be as flexible and general purpose as possible. The process of refinement is divided into basic units and an independent computer program handles each task. Each functional unit communicates with other programs in the package by way of files of well defined format. To modify or replace any program, the user need only understand the function of that particular element. Stereochemical restraints are defined in a general way that can be applied to proteins, nucleic acids, prosthetic groups, solvent atoms and so on. Guide values for bond lengths and bond angles are specified in a straightforward direct manner. Designated groups of atoms can be held constant or constrained to behave as a rigid body during refinement. In order to make the package as efficient as possible, the fast Fourier transform algorithm is used for all the crystallographic transformations. To highlight potential errors in the refined structure the user can list those atoms that have the worst bond lengths and angles, or have the largest positional, temperature-factor or occupancy gradients. It is also possible to check that protein and solvent atoms do not sterically clash with symmetry-related neighbors. Applications of the program package to a bacteriochlorophyll-containing protein, thermolysin-inhibitor complexes and mutants of bacteriophage T4 lysozyme are described.

1 Introduction

There are a number of potential difficulties with the refinement of macromolecular structures including the unfavorable ratio of observations to parameters, the magnitude of the computational requirements, and deficiencies in the starting model ranging from small errors in the coordinates to gross errors arising from misinterpretation of the electron density map. These difficulties have led to the development of different refinement strategies, each of which has its own advantages and disadvantages (*e.g.* Diamond, 1971; Watenpaugh, Sieker, Herriott & Jensen, 1973; Freer, Alden, Carter & Kraut, 1975; Sussman, Holbrook, Church & Kim, 1977; Jack & Levitt, 1978; Konert & Hendrickson, 1980; Agarwal, 1978; Jones & Liljas, 1984). In order

to increase the number of observations it is usual to include knowledge of the stereochemistry of the protein. Bond distances, bond angles, planarity and limits on the approach distances of non-bonded atoms can all be specified. It can also be profitable to incorporate additional information, as in the co-refinement of bovine pancreatic trypsin inhibitor with X-ray and neutron data (Wlodawer & Hendrickson, 1982). If other data are available, such as independent phase information from isomorphous replacement, anomalous scattering, or phase information from molecular replacement, it might be desirable to include this information as well.

The package of programs described uses the principle of restrained least-squares refinement. The package is designed to be as general purpose as possible. Stereochemistry, for example is defined in a general way that can be applied to proteins, nucleic acids, prosthetic groups, solvent atoms and so on. The package has been made as efficient as possible by using the fast Fourier transform algorithm to carry out all the crystallographic transformations.

One limitation of many refinement programs is their inflexibility; portions cannot be replaced nor new functions added without extensive modification of the existing code. This limits the ability of the user to experiment with different refinement strategies. In order to modify the refinement program one must understand the data structure and algorithms of the entire program. The refinement package described here was designed to avoid this limitation. The process of refinement is broken down into basic units and an independent computer program handles each task. Each functional unit communicates with the other programs in the package by way of files of well defined format. To modify or replace any program only requires that the user understand the function of that program; the rest of the programs will function as before. In this manner calculations which can be optimized by space-group-specific algorithms (such as fast Fourier transforms) can be calculated differently for different crystal structures by a simple substitution of the appropriate program.

2 Feature and organization

The package of computer programs was designed to meet five specific objectives which will be briefly discussed below.

1. It should be possible to replace existing functions or add new functions

without modifying existing code.

2. The programs should have a common mechanism for reading data.
3. It should be easy to define standard geometry for new and unusual chemical groups.
4. It should be possible to constrain specified groups of atoms to behave as rigid bodies or to be held constant during refinement.
5. The program should provide tools to aid the user in the detection of errors in the model that are beyond the ability of the refinement package to correct.

Because of the ability to replace easily the programs performing specific calculations, the most efficient algorithms for a particular problem can be used. This feature has resulted in a package with great flexibility and speed.

2.1 Overall organization

The need to partition a refinement program into independent functional units was mentioned in the *Introduction* and has shaped the overall organization and structure of the package. The different functions that are minimized in the refinement are treated as separate ‘terms’ where each term is defined on the basis of the calculations required to evaluate the term and its gradient. Most commonly, two terms are included: a crystallographic and a stereochemical term. The programs required to calculate a term and its gradient are collectively referred to as a ‘module’. The overall refinement package consists of the control program plus a variable number of modules. The control program combines the information presented by all the modules to determine the direction in which to shift the parameters of the model, and, eventually, to determine the magnitude of the shift. This program ‘knows’ nothing about the specifics of the terms that are handled by the various modules. The stereochemical module is implemented as a single program. Because it involves several Fourier transforms, the crystallographic module is broken up into five different programs. This fractionation allows appropriate space-group-specific fast Fourier transform (FFT) programs to be used for different projects.

2.2 Data input and data transfer

Because of the number and independence of the programs in the package it is very desirable that the input format for all programs be the same. Although some data file are created by the computer and others by the user, the style of input has been designed with emphasis on the benefit of the user. All input is token based. The input is read in free format, each token being separated from the next by a delimiter such as a space or a comma (see Tables 1 and 2 for examples). The first token on the card is the keyword. Keywords fall into two classes: data keywords and command keywords. The order of data cards is unimportant. When a command card is encountered, the required operation is performed on whatever data have been read to that point. Input and output is handled by a set of library routines which perform the basic operations of reading in cards, separating tokens, and building numbers from particular tokens.

A related objective was to allow the constituent programs to be easy to write and understand. Sets of common routines have been placed in libraries that are used by most of the programs. These libraries contain routines that process data cards and build the internal data structures. Also there are other routines that locate required information within the data structures. By not having to rewrite these routines one can implement a new program in a very short time and, by having a common internal structure, the existing programs can be understood more easily. On the other hand, although these library routines are available, they do not have to be used when one wishes to add a new program to the refinement package.

2.3 Geometry definition

Because one often needs to include unusual inhibitors or cofactors in the refinement, it is very desirable that the definition of geometry should be general. Often the structures of these small molecules have not been determined and their 'ideal' geometry must be constructed from the fragments whose structures are known. In the present package, standard geometry is defined by breaking the structural model into components, such as amino acids, nucleotides or cofactors. The geometry restraints are then defined in a general way for each component and for the linkages between components.

There are two ways in which stereochemical information can be incorpo-

Table 1: Definition of the general data cards

These are the definitions for the cards used to define the standard geometry for a molecular model. Each atom card contains the name, type and coordinates of the atom as well as the names of the residue and chain in which it resides. Each chain has its type defined on a CHAIN card. The sequence and connectivity of that type of chain is defined on several RESIDUE cards and the restraints associated with each residue type and linkage type are defined on GEOMETRY cards. Table 2 gives a specific example.

The nomenclature is :

<name> is a word or number
 A|B means A or B
 {A} means A is repeated 0 or more times
 “|” means that the | is to be taken literally

Data Cards:

<Atom card> ::=
 <Atom Keyword> <Atom type> <Atom parameters>
 <Atom name> <Residue name> <Chain name>

<Chain card> ::=
 CHAIN <Chain name> <Chain type>
 {<Chain name> “|” <Residue name> <Linkage type>}

<Residue card> ::=
 RESIDUE <Chain name> “|” <Residue name> <Residue type>
 {<Residue name> <Linkage type>}

<Geometry card> ::=
 GEOMETRY <Cluster type> <Restraint type> <Standard value>
 <Value’s sigma> <Atom name> {<Atom name>}

where

<Atom keyword> ::= ATOM | ATOMC | ATOMG
 <Cluster type> ::= <Residue type> | <Linkage type>
 <Restraint type> ::= BOND | ANGLE | TORSION
 TRIGONAL | PLANE | CHIRAL

Table 2: Definition of the structure of Cro repressor

First in the example comes the CHAIN cards which define that all four chains O, A, B, and C are of type CRO. Next come the cards (not all of which are shown) which define the meaning of type CRO. A type is given for each amino acid in CRO along with the targets and types of any linkages between this residue and other residues. The residue types and linkage types are defined by obtaining the restraints associated with each through the geometry restraint library. The GEOMETRY cards are representative examples from the deposited restraint library.¹

```
CHAIN O CRO
CHAIN A CRO
CHAIN B CRO
CHAIN C CRO
```

```
RESIDUE CRO|1 MET 2 PEPTIDE
RESIDUE CRO|2 GLU 3 PEPTIDE
RESIDUE CRO|3 GLN 4 PEPTIDE
RESIDUE CRO|4 ARG 5 PEPTIDE
RESIDUE CRO|5 ILE 6 PEPTIDE
```

```
GEOMETRY PEPTIDE BOND 1.45 0.02 N, CA
GEOMETRY PEPTIDE ANGLE 112 3 N, CA, C
GEOMETRY PEPTIDE PLANE 5 0.02 C, CA, O,+N,+CA
GEOMETRY PEPTIDE TORS 2180 10 CA,C,+N,+CA
GEOMETRY MET BOND 1.81 0.02 CG,SD
GEOMETRY MET ANGLE 100.4 3 CG,SD,CE
```

rated into the refinement process; the information can be added as additional observations (restraints) (*cf.* Hermans & McQueen, 1974; Konnert & Hendrickson, 1980) or the model can be parameterized (constrained) in such a way that the stereochemistry is always ‘ideal’ (*cf.* Diamond, 1971; Warne, Gō & Scheraga, 1972). Although the use of constraints rather than restraints does result in a more favorable ratio of observations or parameters, we have chosen the latter approach. There are several reasons for this, the first being that a restrained model is physically more realistic than a constrained one, as constraints are usually implemented (*e.g.* Ten Eyck, Weaver & Matthews, 1976). A second reason is that different types of restraint can be individually weighted. This allows the user to put a smaller weight on geometric restraints for which the ‘ideal’ values are uncertain. Another reason is that the significance of deviations from ideality can be evaluated by calculating the standard deviation of all the observations within the same class of restraint. At the same time the library of standard values can be tested for accuracy. If there is an error, then there will be a systematic difference between the library value and the value obtained from the refined model. (The ‘stereochemistry’ program in the refinement package will perform both the above tests.) Finally, the use of restraints allows the stereochemical information to be incorporated into the refinement in a manner formally similar to the incorporation of the crystallographic observations (see below). This allows simpler and more efficient code.

2.4 Rigid-body refinement

It is often desirable to have the option of holding portions of the structure fixed, or constraining a group to move as a rigid body (*e.g.* Sussman, Holbrook, Church & Kim, 1977). Because these options do not depend on the nature of the function being minimized they have been implemented in the control program.

The control program also allows one to limit the range of values that the temperature factors and occupancies can assume and, in addition, to constrain a group of atoms to have the same temperature factor or occupancy (*e.g.* to allow for partial occupancy by an inhibitor).

2.5 Detection of errors in the model

Finally, in order to highlight potential errors in the current model of the structure, each module can list those atoms that most seriously violate the restraints of the refinement. The stereochemistry module lists, for each class of geometrical restraint, the worst discrepancies between the model and the ‘ideal’ values. Similarly, the crystallographic module can list those atoms with the largest positional, temperature-factor or occupancy gradient. Experience has shown that these lists are particularly helpful in pointing out areas of the model that are likely to be in error and may need to be corrected manually.

3 Theoretical background

The goal is to minimize a suitable function of the observations in terms of a structural model specified by variables such as coordinates, thermal factors and occupancies. The function used in least-squares refinement is

$$M = \sum_j W(j)[Q_o(j) - Q_c(j, \mathbf{p})]^2, \quad (1)$$

where $Q_o(j)$ is the experimental value for the observation j , $Q_c(j, \mathbf{p})$ is the corresponding value calculated from the coordinate and thermal parameters \mathbf{p} that specify the structural model, and $W(j)$ is the desired weighting function. The sum in (1) is over all observations, but can be separated into different terms based, for example, on the crystallographic observations \mathbf{s} and the stereochemical observations b (see Appendix B for additional details):

$$M = \sum_{\mathbf{s}} W(\mathbf{s})[Q_o(\mathbf{s}) - Q_c(\mathbf{s}, \mathbf{p})]^2 + \sum_b W(b)[Q_o(b) - Q_c(b, \mathbf{p})]^2 \quad (2)$$

More terms could be added if other classes of observation were available. The gradient of M can also be separated into similar terms. This means that the calculations for the crystallographic term can be kept completely separate from calculations for the other terms.

The computational problem is to determine a set of parameters which minimized M . There exist function minimization methods which use no derivatives, which use only first derivatives, and which use second derivatives,

in order of increasing power of convergence and increasing computational cost. In the present case there are several reasons for using first-derivative methods.

1. The radius of convergence of first-derivative methods is larger than that of second-derivative methods, and in these problems one often starts far from the minimum.
2. The computational cost of first-derivative methods is proportional to N (the number of parameters) instead of N^2 . For large N this is very important.
3. Implementation of parameter constraints for holding variables constant, or for requiring variables to behave as rigid groups, is particularly simple for first-derivative methods (see below).

In order to hold a parameter constant, one simply sets the derivative of this parameter equal to zero before calculating the parameter shifts. This prevents the corresponding parameters from changing. To treat a set of atoms \mathbf{x}_i as a rigid group one redefines these atoms in terms of a chosen origin \mathbf{x}_o and three orientation parameters α , *i.e.*

$$\mathbf{x}_i = g(\mathbf{x}_o, \alpha). \quad (3)$$

Similarly, the residual M is redefined as

$$M(\mathbf{x}_i) = f(\mathbf{x}_o, \alpha). \quad (4)$$

Then by the chain rule

$$\partial M / \partial \mathbf{x}_i = (\partial f / \partial \mathbf{x}_o)(\partial \mathbf{x}_o / \partial \mathbf{x}_i) + (\partial f / \partial \alpha)(\partial \alpha / \partial \mathbf{x}_i), \quad (5)$$

which gives an overdetermined system of equations for the derivatives of f . This system of equation can be solved by least squares and the solution used to calculate values of $\partial M / \partial \mathbf{x}_i$ consistent with the rigid-body constraint. This had the virtue of always operating in the original parameter space, but has the fault that nonlinearities can distort the rigid group. A more correct method is to perform the parameter shift steps in the (\mathbf{x}_o, α) space and expand to the original space for all other calculations.

First-derivative methods all use the same general strategy, namely calculation of the shift direction followed by a line search for a minimum in the chosen direction. The present package uses the conjugate gradient method (Fletcher & Reeves, 1964). In this procedure the changes in the gradient vector from cycle to cycle are used to approximate the second derivative without actually having to compute this quantity.

By using the method of Agarwal (1978) the amount of computer time required to calculate the gradient of the crystallographic term is only slightly longer than the calculation of a FFT of the structure. The time to calculate the gradient of the stereochemical term is, in comparison, miniscule. The stereochemical and crystallographic gradients are combined with the shift vector of the previous cycle to give the direction (but not the magnitude) of the shift for each parameter. The search along the shift vector for the optimum shift magnitude requires at least three calculations of M , *i.e.* three FFT's plus some additional calculations. Thus the overall computer time required for a single cycle of refinement is approximately four times that required for one FFT. It is apparent that space-group-specific FFT's can substantially reduce the required computer time per cycle. Included in the refinement package is a program (to be described elsewhere) that will calculate space-group-specific FFT's for most noncentrosymmetric space groups.

4 Crystallographic term

The function that we have chosen to minimize is

$$M = \sum_{\mathbf{s}} W(\mathbf{s}) [k|F_o(\mathbf{s})| - |F_c(\mathbf{s}, \mathbf{p})|]^2, \quad (6)$$

where F_o and F_c are the observed and calculated structure factors and k is a scale factor. Note that no exponential factor is applied to the F_c 's. This will force the thermal factors of the individual atoms to include any 'overall' mismatch between the observed and calculated data sets.

At the beginning of each cycle of refinement the scale factor k is determined by minimizing

$$M(k, B) = \sum_{\mathbf{s}} W(\mathbf{s}) [k|F_o(\mathbf{s})| - \exp(-Bs^s/4)|F_c(\mathbf{s}, \mathbf{p})|]^2 \quad (7)$$

where $s = \sin \theta / \lambda$ and $F_c(\mathbf{s})$ is treated as a constant. Equation (7) includes an overall thermal factor B , which is necessary to allow for an initial overall discrepancy between the F_o 's and F_c 's. Although both k and B are treated as variables in the minimization of (7), only k is substituted in (6). As the refinement proceeds, the discrepancy represented by B is absorbed within the thermal factors of the individual atoms and, during successive cycles, rapidly approaches zero.

Each module of the refinement package is able to calculate both the value and the gradient of its term. For the crystallographic term the structure factors are calculated by a space-group-specific FFT (Ten Eyck, 1977). The gradients are calculated by a modified version of the procedure outlined by Agarwal (1978). This modified version was devised by A. Lifchitz (Agarwal, Lifchitz & Dodson, 1981; Isaacs, 1982) and is described in detail in Appendix A. In outline, the procedure is as follows. An $(F_o - F_c)$ map is calculated for the molecular volume. For each parameter in the model a convolution, evaluated at the atomic position, is calculated between this map and the derivative of the calculated atomic electron density function for the atom involved. Because the extent of the electron cloud of a single atom is small, the calculation of this convolution is rapid. Usually the calculation of the convolutions takes about a quarter of the time required to calculate the difference map.

5 Stereochemical term

The major goal in the implementation of this part of the package was to make it as easy as possible for the user to specify 'ideal' bond lengths and angles. Stereochemical restraints are usually introduced either as energy terms (*e.g.* Jack & Levitt, 1978) or by expressing all types of stereochemical restraints as distances (*e.g.* Ten Eyck, Weaver & Matthews, 1976; Dodson, Isaacs & Rollett, 1976; Konnert & Hendrickson, 1980). There are many drawbacks to both approaches. In the first case it may be difficult to obtain reliable energy parameters, especially for novel chemical groups. Also the introduction of an inappropriate energy term might mask interesting and unexpected features of the structure. On the other hand, if one attempts to define standard geometry in terms of interatomic distances, then such distances must be determined indirectly from a known example with ideal geometry. There

are obvious difficulties if no known structure exists for the chemical group in question. In addition, restraints on interatomic distances as a means of specifying bond angles can lead to very distorted planarity of trigonal atoms and aromatic rings.

The method used in this package is to include stereochemical restraints as ‘observations’ but to specify such restraints in a form that is most convenient for the user, *i.e.* as bond lengths, bond angles and so on. There are six classes of stereochemical information with which the structural model can be restrained; bond lengths, bond angles, torsion angles, trigonal planarity, general planarity and contacts between non-bonded atoms. (Chirality is monitored but not restrained because the chirality function is discontinuous and has no derivatives.) Because the program deals directly with the stereochemical information, some of the derivatives are difficult to derive, and, for the planarity restraints, certain assumptions were used to simplify the calculation. The derivations of the gradients for the stereochemical term are given in Appendix B.

To apply the stereochemical restraints the contents of the asymmetric unit are broken up into different hierarchical units. Each unit can be broken up into small subgroups of atoms in whatever manner is appropriate for the problem at hand. For example, consider the crystal structure of Cro repressor (Anderson, Ohlendorf, Takeda & Matthews, 1981). The asymmetric unit consists of four chemically identical polypeptides, each with 66 amino acids. The first hierarchical unit is defined by CHAIN cards. In this example we specify that there are four chains, named O, A, B and C, each chain being of type ‘CRO’ (see Table 2 for representative data cards). The makeup of a ‘CRO’ chain is then defined by RESIDUE cards. A series of such cards is used to define the sequence of units in the chain (in this case, amino-acid residues) and the types of linkages between successive units (in this case peptide bonds). In this example, the units of the chain are named GLY, ALA, THR, ... *etc.*, and the linkages PEPTIDE, SS, ... *etc.* The geometric restraints associated with each unit or linkage type are defined with GEOMETRY cards. Each restraint (bond length, bond angle, torsion angle, plane, ...) is specified in a straightforward manner. There is no particular order in which these cards must be given and they can be arranged into different files in any desired manner.

The enumeration of all the stereochemical restraints in this manner may seem to be time consuming, but most of the files, once created, can be trans-

ferred from one application to another. Also it is easy to inspect and alter the ideal values of the restraints since they appear in the program in the same form as in everyday usage. A table which gives the library of 'ideal' stereochemistry that has been adopted in this laboratory, primarily from Bowen, Donohue, Jenkin, Kennard, Wheatley & Whiffen (1958) and Vijayan (1976), has been deposited.²

Interactions between non-bonded atoms cannot be defined in the manner described above because one does not know in advance which atoms may approach each other. Close contacts are discovered by generating a list of all pairs of atoms which are closer to each other than specified values and discarding from consideration any pairs which are bonded, or and involved in 1-3 or 1-4 type contacts. The 1-3 and 1-4 contacts are better dealt with as bond angles and torsion angles. The standard value for the closest distance allowed before any action is taken is defined in terms of the elemental types of the two atoms. This method of definition allows a closer approach between atoms which have the potential of forming a hydrogen bond or a salt bridge than the distance allowed for atoms in van der Waals contact. The program will prevent non-bonded atoms from moving too close together but no attractive force is applied to atoms that are beyond the specified approach distance.

One novel feature of the program is the ability to avoid steric clashes between adjacent molecules in the crystal. By specifying the appropriate symmetry operators the list of potential non-bonded contacts can be extended to include molecules that surround the reference structure. This procedure is particularly useful in avoiding 'duplicate' or 'overlapping' solvent atoms.

The program that implements the stereochemistry module has a number of additional features. It can list the worst discrepancies in the model for each type of geometry restraint and provide overall statistics for each class. Also it can produce a table which compares the 'ideal' value of each restraint with the average value in the present model. This table is useful when looking for potential errors in the geometry library.

²The geometry restraint library has been deposited with the British Library Document Supply Centre as Supplementary Publication No. SUP 43532 (8 pp.). Copies may be obtained through The Executive Secretary, International Union of Crystallography, 5 Abbey Square, Chester CH1 2HU, England.

6 The control program

The control program has two major functions: (1) to determine the overall direction of shift for each parameter, and (2) to determine the optimum magnitude (fraction of the shift) to be applied along the shift vector.

In an initial cycle of refinement, the overall gradient is obtained by combining the contributions from the crystallographic, stereochemical and any other terms and the direction of shift is obtained by the method of steepest descent. For second and subsequent cycles of refinement one can combine the overall gradient with the direction of the shift used in the previous cycle and determine the new shift direction by the conjugate-gradient technique. The control program also determines the optimal shift magnitude by searching along the shift direction as described previously.

A number of options exist for modifying the shift vector before the shift is applied. One can combine all or part of the structure into units within which all the atoms are treated identically (*e.g.* Table 3). If the atoms within an amino acid are combined, the temperature-factor shift applied to each atom will be the average of the individual shifts. Positional parameters can be treated as though the group were a rigid body. This is done by a least-squares fitting procedure. It is also possible to place upper and lower limits on the values of the thermal factors and the occupancies.

7 Applications

The package of programs has been in productive use (while being developed) for four years. It had been applied to a number of different refinement tasks in this laboratory and elsewhere. Up to the present it has only been used on VAX/VMS systems. In this section we briefly review some of the applications.

7.1 Bacteriochlorophyll *a* protein

The first extensive use of the package was in the refinement of the bacteriochlorophyll *a* protein (Tronrud, Schmid & Matthews, 1986). This molecule (molecular weight 150 000 daltons) consists of three identical subunits related by a threefold axis of symmetry. Each of the subunits consists of a

Table 3: Examples of parameter editing cards

1. Specification of rigid-body refinement

```
COMBINE XYZ 360 - 361
COMBINE XYZ 362 - 363
COMBINE XYZ 364 - 365
COMBINE XYZ 366 - 367
COMBINE XYZ 367 - 369
COMBINE XYZ 370 - 371
COMBINE XYZ 372 - 373
```

2. Refinement of the occupancy of a bound inhibitor

```
FIX OCC 1 - COOH
FIX OCC SOL1
FIX OCC ZINC:ZINC
FIX OCC CAL1:CAL
FIX OCC CAL2:CAL
FIX OCC CAL3:CAL
FIX OCC CAL4:CAL
COMBINE OCC INHIBITOR
FIX B
FIX XYZ
```

Part (1) shows the cards required to command the control program to treat the seven bacteriochlorophyll *a* molecules as seven rigid bodies. Each molecule contains two residues: the ring and the phytol tail. The first card is read as ‘combine the positional parameters for all residues between residue 360 and 361’. The operation ‘all residues between’ is defined as taking the first residue and then following the linkage definitions in the RESIDUE cards including all residues until the last residue is found.

Part (2) shows the cards required to refine a single occupancy parameter for an inhibitor of the protein thermolysin. The first card fixed the occupancy of all of the protein atoms. COOH is the residue name used in this project for the extra oxygen at the carboxyl terminus. The next card fixes the occupancy of all of the solvent atoms. They are defined in a single residue to make operations such as this simple. Then the occupancies for the zinc atom and the four calcium atoms are fixed. Finally the occupancy parameters for the inhibitor, presumed to be defined as a single residue called INHIBITOR, are combined. The last two cards simply state that the positional and thermal parameters for all atoms in the model should be held constant.

Table 4: Representative refinements of macromolecular structures

Protein	Bchl protein	Thermolysin*	Thermolysin [†]	T4 Lysozyme [‡]
Space group	$P6_3$	$P6_122$	$P6_122$	$P3_221$
Cell Dimensions				
a, b (Å)	111.9	94.1	94.1	61.2
c (Å)	98.6	131.4	131.4	97.4
Number of atoms	3086	2637	2643	1429
Resolution (Å)	1.9	2.3	1.6	1.7
Number of reflections	43 589	13 523	31 627	15 791
Final R value	18.9%	17.4%	17.7%	17.4%
Deviations from ideal values:				
bond lengths (Å)	0.02	0.02	0.02	0.02
bond angles (°)	3.2	2.9	2.8	2.6
Cycles of refinement	120 [§]	16	28	29
Time per cycle (c.p.u. hours)	2.0	0.8	1.5	0.75

*Thermolysin:phosphoramidon-inhibitor complexes (Tronrud, Monzingo & Matthews, 1986).

[†]Complex of thermolysin with carbobenzoxy-Gly^P(OH)-L-Leu-L-Leu (Tronrud, Holden & Matthews, 1987).

[‡]Bacteriophage T4 lysozyme mutant Thr 157 → Glu (unpublished results of Alber, Wilson, Matthews *et al.*).

[§]Includes cycles used to test and debug various components of the refinement packages.

polypeptide chain of approximately 350 amino acids that enclose seven bacteriochlorophyll *a* molecules (Fenna & Matthews, 1975; Matthews, Fenna, Bolognesi, Schmid & Olson, 1979). During the course of the refinement the amino-acid sequence of the protein was not known, but has been reported subsequently (Daurat-Larroque, Brew & Fenna, 1986).

There were several factors that led to the adoption of the present refinement package. The first was the size of the computation problem. As summarized in Table 4, the asymmetric unit contains 3086 atoms and there are 43 598 reflections to 1.9Å resolution. In addition, the space group ($P6_3$) precludes full use of the crystallographic symmetry to reduce the size of the FFT calculations. Experience with other projects in the laboratory suggests that the Hendrickson refinement program (Hendrickson & Konnert, 1980) would require about 48h of c.p.u. time per refinement cycle on our VAX 11/780

(subsequent improvements to the Hendrickson program have substantially improved its computational efficiency). We had also had experience in the laboratory with EREF (Jack & Levitt, 1978). This program is substantially faster, on a per cycle basis, than the Hendrickson program. A disadvantage of using EREF for the refinement of the Bchl protein arose from the presence of the Bchl rings. Because of the diversity of bond lengths and bond angles in the seven bacteriochlorophylls, as well as uncertainties in their energetics, the definition of standard geometry for use by EREF appeared to be quite difficult.

As discussed previously, the definition of standard geometry in the present refinement package is very flexible, and readily adaptable to ‘unusual’ situations. In particular, in the present situation it was not necessary to force all the conjugated atoms in the Bchl rings to lie in a single plane. Rather, we divided the conjugated atoms into appropriate sets of overlapping sub-planes (Tronrud, Schmid & Matthews, 1986). This method of restraint maintained local planarity, but allowed larger-scale deformations. This procedure led to the finding that the seven Bchl rings exhibit two distinct classes of bending, one of which is also observed in the structure of ethyl chlorophyllide *a* (Tronrud, Schmid & Matthews, 1986). An inappropriate application of restraints to the Bchl rings could well have masked this small but significant effect.

In the initial refinement of the Bchl protein the refinement package used analytical summations to calculate the gradient of the crystallographic term. With this method each cycle of refinement required 8h of c.p.u. time on our VAX 11/780. The present version of the program requires 2.0h for the same problem.

The general strategy of refinement that we have adopted for the Bchl and other proteins is first to refine for several cycles with weak geometry restraints, then to run a few cycles of temperature-factor refinement, then restore the model to good stereochemistry by refining for several cycles with strong geometry restraints and finish with several additional cycles of thermal factor refinement. At this stage the resulting difference electron density map and ‘ $2F_o - F_c$ ’ map are inspected on the graphics system in the usual way. Potential problem areas are highlighted by inspecting the lists of worst bond lengths, bond angles, departures from possible errors in the assumed amino-acid sequence. In this instance the cycles of refinement followed by inspection of the model were repeated seven times to achieve the final refined

structure and 'X-ray' amino acid sequence. The overall refinement statistics are summarized in Table 4.

7.2 Thermolysin-inhibitor complexes

The refinement package has been used extensively to study a number of complexes of inhibitors with the thermostable endopeptidase thermolysin (*e.g.* Holmes, Tronrud & Matthews, 1983; Tronrud, Monzingo & Matthews, 1986). In different instances the resolution ranges from 2.3 to 1.6Å.

Because of the hexagonal space group and the size of the problem (Table 2), calculations of structure factors and crystallographic derivatives by conventional methods is time consuming. [Refinement of the native structure at 1.6Å resolution by the method of Hendrickson & Konnert (1980) required 21 c.p.u. hours per cycle (Holmes & Matthews, 1982). The present program package requires 0.7 to 1.5h per cycle, depending on the resolution. The ability of the present program package to specify the geometry of chemically unusual inhibitors is an advantage. In addition, it is also possible to define appropriate stereochemistry for inhibitors that are covalently bonded to the enzyme (*e.g.* Holmes, Tronrud & Matthews, 1983).

The refinement of an inhibitor complex normally requires 10 - 30 cycles of refinement. Little manual intervention is required because thermolysin normally does not change its conformation very much when inhibitors are bound. It is, of course, always necessary to monitor the configuration of the inhibitor during refinement and to check for changes in solvent structure concomitant with inhibitor binding.

7.3 Bacteriophage T4 mutant structures

As part of a program to determine the roles of individual amino acids in stabilizing protein structures, the structures of a series of mutant T4 phage lysozymes have been determined (*e.g.* Grütter, Weaver, Gray & Matthews, 1983; Alber, Grütter, Gray, Wozniak, Weaver, Chen, Baker & Matthews, 1986). The refinement of each mutant (unpublished results) is in principle very similar to the refinement of an enzyme-inhibitor complexes. One starts with the refined structure of the native protein, locally modified to correspond to the structure of the mutant. In the early stages of the refinement the stereochemical restraints are kept weak to allow the starting model to relax

to conform to the diffraction data observed for the mutant structure. When there appear to be no systematic shifts in the coordinates from one refinement cycles to the next, the stereochemical restraints are strengthened in order to enforce the ‘ideal’ geometry. Some representative refinement statistics are summarized in Table 4. Experience with refining these mutant structures at resolutions in the range 1.9 - 1.7Å suggests that the refinement procedure is capable of successfully moving both main-chain and side-chain atoms through distances of 1.0Å.

Several of our colleagues have contributed to the development of the refinement package by applying it to their own problems and suggesting modifications and improvements; in particular, we thank Drs M. A. Holmes and M. F. Schmid. Dr Holmes also suggested ‘TNT’ as the name for the package. We also thank Dr N. Isaacs for providing detailed information on the Lifchitz modification of the Agarwal fast Fourier transform algorithm. The work was supported in part by grants from the national Institutes of Health (GM 20066; GM 21967; GM 35114), the National Science Foundation (DMB 8611084) and the Murdock Charitable Trust.

A Efficient calculation of gradients by means of the fast Fourier transform algorithm

By Dale E. Tronrud

A.1 Definitions

The following symbols are used in both the main paper and the Appendices.

\mathbf{p}	The vector consisting of all refinable parameters in the model.
\mathbf{p}_i	The vector of all parameters of atom i .
\mathbf{x}	The vector of all positional parameters in the model.
\mathbf{x}_i	The vector of the positional parameters of atom i .
x_i, y_i, z_i	The coordinates of atom i .
B_i	The thermal factor for atom i .
O_i	The occupancy of atom i .
f_i	The atomic scattering factor for atom i .
\mathbf{s}	The vector (h, k, l) ; $s = \sin \theta / \lambda$.
$Q_o(j)$	The observed value for observation j .
$Q_c(j, \mathbf{p})$	The value calculated for observation j from the parameters \mathbf{p} .
$W(j)$	A weighting factor for observation j .
$F_o(\mathbf{s})$	The observed structure factor for reflection \mathbf{s} .
$F_c(\mathbf{s}, \mathbf{p})$	The structure factor for reflection \mathbf{s} calculated from the parameter \mathbf{p} .
T	A Fourier transform.
T^{-1}	An inverse Fourier transform.
$*$	A convolution.
$\left(\frac{*}{\mathbf{x}_i}\right)$	The value of the convolution evaluated at \mathbf{x}_i .

A.2 Lifchitz variation of the FFT algorithm

To refine a structural model against crystallographic data both the function (8) and its gradient (9) must be evaluated.

$$M(\mathbf{p}) = \sum_{\mathbf{s}} W(\mathbf{s}) [k|F_o(\mathbf{s})| - |F_c(\mathbf{s}, \mathbf{p})|]^2 \quad (8)$$

$$dM(\mathbf{p})/d\mathbf{p} = -2 \sum_{\mathbf{s}} W(\mathbf{s}) [k|F_o(\mathbf{s})| - |F_c(\mathbf{s}, \mathbf{p})|] d|F_c(\mathbf{s}, \mathbf{p})|/d\mathbf{p}. \quad (9)$$

Because the evaluation of the gradient involves a sum over all reflections for each parameter of the model, the calculation would be very time consuming if performed as in (9).

Agarwal (1978) showed that (9) can be expressed as follows:

$$\partial M(\mathbf{p})/\partial x_i = T^{-1} \{2W(\mathbf{s}) [k|F_o(\mathbf{s})| - |F_c(\mathbf{s}, \mathbf{p})|] \exp[i\varphi_c(\mathbf{s}, \mathbf{p})] (2\pi i h) g_i(\mathbf{s})\} \quad (10)$$

$$\partial M(\mathbf{p})/\partial y_i = T^{-1} \{2W(\mathbf{s}) [k|F_o(\mathbf{s})| - |F_c(\mathbf{s}, \mathbf{p})|] \exp[i\varphi_c(\mathbf{s}, \mathbf{p})] (2\pi i k) g_i(\mathbf{s})\} \quad (11)$$

$$\partial M(\mathbf{p})/\partial z_i = T^{-1}\{2W(\mathbf{s})[k|F_o(\mathbf{s})| - |F_c(\mathbf{s}, \mathbf{p})|] \exp[i\varphi_c(\mathbf{s}, \mathbf{p})](2\pi il)g_i(\mathbf{s})\} \quad (12)$$

$$\partial M(\mathbf{p})/\partial B_i = T^{-1}\{-2W(\mathbf{s})[k|F_o(\mathbf{s})| - |F_c(\mathbf{s}, \mathbf{p})|] \exp[i\varphi_c(\mathbf{s}, \mathbf{p})](-s^2/4)g_i(\mathbf{s})\} \quad (13)$$

where

$$g_i(\mathbf{s}) = O_i f_i(\mathbf{s}) \exp(B_i s^2/4). \quad (14)$$

This formulation requires a Fourier transform for each parameter and is also impractical. However, Agarwal separated the portion of (10)-(13) that depend on the atom (subscript i) from the rest of the factors and applied the convolution theorem to arrive at the following equations:

$$\partial M(\mathbf{p})/\partial x_i = T^{-1}\{W(\mathbf{s})[k|F_o(\mathbf{s})| - |F_c(\mathbf{s}, \mathbf{p})|] \exp[i\varphi_c(\mathbf{s}, \mathbf{p})](2\pi ih)\} \left(\frac{*}{\mathbf{x}_i}\right) T^{-1}[2g_i(\mathbf{s})] \quad (15)$$

$$\partial M(\mathbf{p})/\partial y_i = T^{-1}\{W(\mathbf{s})[k|F_o(\mathbf{s})| - |F_c(\mathbf{s}, \mathbf{p})|] \exp[i\varphi_c(\mathbf{s}, \mathbf{p})](2\pi ik)\} \left(\frac{*}{\mathbf{x}_i}\right) T^{-1}[2g_i(\mathbf{s})] \quad (16)$$

$$\partial M(\mathbf{p})/\partial z_i = T^{-1}\{W(\mathbf{s})[k|F_o(\mathbf{s})| - |F_c(\mathbf{s}, \mathbf{p})|] \exp[i\varphi_c(\mathbf{s}, \mathbf{p})](2\pi il)\} \left(\frac{*}{\mathbf{x}_i}\right) T^{-1}[2g_i(\mathbf{s})] \quad (17)$$

$$\partial M(\mathbf{p})/\partial B_i = T^{-1}\{W(\mathbf{s})[k|F_o(\mathbf{s})| - |F_c(\mathbf{s}, \mathbf{p})|] \exp[i\varphi_c(\mathbf{s}, \mathbf{p})](-s^2/4)\} \left(\frac{*}{\mathbf{x}_i}\right) T^{-1}[-2g_i(\mathbf{s})]. \quad (18)$$

If the atomic scattering factor is modeled as a sum of Gaussians the second Fourier transform can be calculated analytically. For a given model, the first transform in each of (15)-(18) can be calculated with the FFT algorithm. In the form given in (15)-(18), it is necessary to perform three Fourier transform to determine the positional derivatives.

Lifchits (see Agarwal, Lifchitz & Dodson, 1981) has pointed out that the gradient can be calculate more efficiently by factoring (10)-(13) so that the first Fourier transform is the same in all the equations. The refactoring results in the following equations:

$$\partial M(\mathbf{p})/\partial x_i = T^{-1}\{W(\mathbf{s})[k|F_o(\mathbf{s})| - |F_c(\mathbf{s}, \mathbf{p})|] \exp[i\varphi_c(\mathbf{s}, \mathbf{p})]\} \left(\frac{*}{\mathbf{x}_i}\right) T^{-1}[2(2\pi ih)g_i(\mathbf{s})] \quad (19)$$

$$\partial M(\mathbf{p})/\partial y_i = T^{-1}\{W(\mathbf{s})[k|F_o(\mathbf{s})| - |F_c(\mathbf{s}, \mathbf{p})|] \exp[i\varphi_c(\mathbf{s}, \mathbf{p})]\} \left(\frac{*}{\mathbf{x}_i}\right) T^{-1}[2(2\pi ik)g_i(\mathbf{s})] \quad (20)$$

$$\partial M(\mathbf{p})/\partial z_i = T^{-1}\{W(\mathbf{s})[k|F_o(\mathbf{s})|-|F_c(\mathbf{s}, \mathbf{p})|] \exp[i\varphi_c(\mathbf{s}, \mathbf{p})]\} \left(\frac{*}{\mathbf{x}_i}\right) T^{-1}[2(2\pi il)g_i(\mathbf{s})] \quad (21)$$

$$\partial M(\mathbf{p})/\partial B_i = T^{-1}\{W(\mathbf{s})[k|F_o(\mathbf{s})|-|F_c(\mathbf{s}, \mathbf{p})|] \exp[i\varphi_c(\mathbf{s}, \mathbf{p})]\} \left(\frac{*}{\mathbf{x}_i}\right) T^{-1}[-2(-s^2/4)g_i(\mathbf{s})]. \quad (22)$$

When these equations are used to calculate the gradient of $M(\mathbf{p})$, only a single FFT is required. If one remembers that $g_i(\mathbf{s})$ is simply a sum of Gaussians, the Fourier transforms on the right can be determined analytically by using the following rules:

$$T^{-1}[\exp(-Bs^2)] = (\pi/B)^{3/2} \exp(-\pi/B r^2) \quad (23)$$

$$T^{-1}[2\pi ihF(\mathbf{s})] = -\partial T^{-1}[F(\mathbf{s})]/\partial x \quad (24)$$

$$T^{-1}[s^2 \exp(-Bs^2)] = (\pi/B)^{3/2} \exp(-\pi^2 r^2/B) [3 - 2(\pi^2/B)r^2] \quad (25)$$

A.3 The reduction of series termination errors

The calculation of the convolutions in (19)-(22) involves sampling each of the functions at discrete points, multiplying the values point by point, and summing all the products. The sampling interval required to represent a function with a given accuracy depends on the magnitude of the high-resolution components of that function. There is no problem determining the sampling interval for the function on the left of the convolution because it contains no components of resolution higher than the measured data. However, the function on the extreme right-hand side of (19)-(22) are not of limited resolution and therefore, at least in principle, must be sampled on a very fine grid. This problem existed in the original equations (15)-(18) but is much more serious with the new form of (19)-(22) because the high-resolution components are enhanced by the inclusion of the crystallographic indices (h, k, l) . This problem is most severe in the calculation of the temperature-factor derivatives (22) because of the s^2 factor. A mechanism to allow these functions to be calculated using the somewhat coarser grid has been devised by recognizing that the errors introduced by a coarse grid are fundamentally the same as those encountered in the calculation of structure factors using the FFT method (Ten Eyck, 1977). The solution involves ‘smearing’ or ‘blurring’ the function of interest so that it is sampled by a larger number of grid points. The ‘smearing’ must be compensated elsewhere in the calculation. In this case the compensation is achieved by ‘sharpening’ the difference map.

This can be done without introducing additional errors because no new high-resolution terms are introduced into the difference map. The final equations, as used in the program ADERIV, are

$$\begin{aligned} \frac{\partial M(\mathbf{p})}{\partial x_i} &= T^{-1}\{W(\mathbf{s})[k|F_o(\mathbf{s})| - |F_c(\mathbf{s}, \mathbf{p})|] \exp[i\varphi_c(\mathbf{s}, \mathbf{p})] \exp(B^0 s^2/4)\} \\ &\stackrel{(*)}{(x_i)} T^{-1}[2(2\pi i h)g_i(\mathbf{s}) \exp(-B^0 s^2/4)] \end{aligned} \quad (26)$$

$$\begin{aligned} \frac{\partial M(\mathbf{p})}{\partial y_i} &= T^{-1}\{W(\mathbf{s})[k|F_o(\mathbf{s})| - |F_c(\mathbf{s}, \mathbf{p})|] \exp[i\varphi_c(\mathbf{s}, \mathbf{p})] \exp(B^0 s^2/4)\} \\ &\stackrel{(*)}{(x_i)} T^{-1}[2(2\pi i k)g_i(\mathbf{s}) \exp(-B^0 s^2/4)] \end{aligned} \quad (27)$$

$$\begin{aligned} \frac{\partial M(\mathbf{p})}{\partial z_i} &= T^{-1}\{W(\mathbf{s})[k|F_o(\mathbf{s})| - |F_c(\mathbf{s}, \mathbf{p})|] \exp[i\varphi_c(\mathbf{s}, \mathbf{p})] \exp(B^0 s^2/4)\} \\ &\stackrel{(*)}{(x_i)} T^{-1}[2(2\pi i l)g_i(\mathbf{s}) \exp(-B^0 s^2/4)] \end{aligned} \quad (28)$$

$$\begin{aligned} \frac{\partial M(\mathbf{p})}{\partial B_i} &= T^{-1}\{W(\mathbf{s})[k|F_o(\mathbf{s})| - |F_c(\mathbf{s}, \mathbf{p})|] \exp[i\varphi_c(\mathbf{s}, \mathbf{p})] \exp(B^0 s^2/4)\} \\ &\stackrel{(*)}{(x_i)} T^{-1}[-2(-s^2/4)g_i(\mathbf{s}) \exp(-B^0 s^2/4)] \end{aligned} \quad (29)$$

A.4 Generalized uses of the algorithm

What was not clear in Agarwal's original paper was that this computational short cut can be used in many cases other than the evaluation of (9). If the derivation is carried out for the general case we discover that the identity (30)-(33) holds whenever $E(\mathbf{s})$ is a symmetric function.

$$\begin{aligned} \sum_{\mathbf{s}} E(\mathbf{s})\partial|F_c(\mathbf{s}, \mathbf{p})|/\partial x_i &= T^{-1}\{E(\mathbf{s}) \exp[i\varphi_c(\mathbf{s}, \mathbf{p})]\} \\ &\stackrel{(*)}{(x_i)} T^{-1}[(2\pi i h)g_i(\mathbf{s})] \end{aligned} \quad (30)$$

$$\begin{aligned} \sum_{\mathbf{s}} E(\mathbf{s})\partial|F_c(\mathbf{s}, \mathbf{p})|/\partial y_i &= T^{-1}\{E(\mathbf{s}) \exp[i\varphi_c(\mathbf{s}, \mathbf{p})]\} \\ &\stackrel{(*)}{(x_i)} T^{-1}[(2\pi i k)g_i(\mathbf{s})] \end{aligned} \quad (31)$$

$$\begin{aligned} \sum_{\mathbf{s}} E(\mathbf{s}) \partial |F_c(\mathbf{s}, \mathbf{p})| / \partial z_i &= T^{-1} \{ E(\mathbf{s}) \exp[i\varphi_c(\mathbf{s}, \mathbf{p})] \} \\ &\stackrel{(*)}{(x_i)} T^{-1} [(2\pi i l) g_i(\mathbf{s})] \end{aligned} \quad (32)$$

$$\begin{aligned} \sum_{\mathbf{s}} E(\mathbf{s}) \partial |F_c(\mathbf{s}, \mathbf{p})| / \partial B_i &= T^{-1} \{ E(\mathbf{s}) \exp[i\varphi_c(\mathbf{s}, \mathbf{p})] \} \\ &\stackrel{(*)}{(x_i)} T^{-1} [(-s^2/4) g_i(\mathbf{s})] \end{aligned} \quad (33)$$

We can use (30)-(33) to speed up the calculation of the gradient of almost any function involving structure factors.

Let us develop an example. Suppose that we wish to minimize not the usual function of the X-ray data (8), but the negative of the correlation coefficient $r(\mathbf{p})$ between the observed and calculated structure factors, which can be cast as in (35), where the bar indicates the mean value.

$$r'(\mathbf{p}) = -r(\mathbf{p}) \quad (34)$$

$$\begin{aligned} &= \frac{[\overline{|F_o(\mathbf{s})| |F_c(\mathbf{s}, \mathbf{p})|} - \overline{|F_o(\mathbf{s})|} \overline{|F_c(\mathbf{s}, \mathbf{p})|}]}{\{[\overline{|F_o(\mathbf{s})|^2} - \overline{|F_o(\mathbf{s})|^2}] [\overline{|F_c(\mathbf{s}, \mathbf{p})|^2} - \overline{|F_c(\mathbf{s}, \mathbf{p})|^2}]\}^{-1/2}}. \end{aligned} \quad (35)$$

The gradient is given by

$$\begin{aligned} \frac{dr'(\mathbf{p})}{d\mathbf{p}} &= \frac{r'(\mathbf{p})}{n} [\overline{|F_o(\mathbf{s})|} \sum_{\mathbf{s}} d|F_c(\mathbf{s}, \mathbf{p})|/d\mathbf{p} \\ &\quad - \sum_{\mathbf{s}} |F_o(\mathbf{s})| d|F_c(\mathbf{s}, \mathbf{p})|/d\mathbf{p}] \\ &\quad \times [\overline{|F_o(\mathbf{s})| |F_c(\mathbf{s}, \mathbf{p})|} - \overline{|F_o(\mathbf{s})|} \overline{|F_c(\mathbf{s}, \mathbf{p})|}]^{-1} \\ &\quad - \{ [\sum_{\mathbf{s}} |F_c(\mathbf{s}, \mathbf{p})| d|F_c(\mathbf{s}, \mathbf{p})|/d\mathbf{p} \\ &\quad - \overline{|F_c(\mathbf{s}, \mathbf{p})|} \sum_{\mathbf{s}} d|F_c(\mathbf{s}, \mathbf{p})|/d\mathbf{p}] \\ &\quad \times [\overline{|F_c(\mathbf{s}, \mathbf{p})|^2} - \overline{|F_c(\mathbf{s}, \mathbf{p})|^2}]^{-1} \}, \end{aligned} \quad (36)$$

where n is the number of structure factors included. To calculate this gradient we need a number of means and three complicated summations, (37), (38) and (39):

$$\sum_{\mathbf{s}} d|F_c(\mathbf{s}, \mathbf{p})|/d\mathbf{p} \quad (37)$$

$$\sum_{\mathbf{s}} |F_o(\mathbf{s})| d|F_c(\mathbf{s}, \mathbf{p})|/d\mathbf{p} \quad (38)$$

$$\sum_{\mathbf{s}} |F_c(\mathbf{s}, \mathbf{p})| d|F_c(\mathbf{s}, \mathbf{p})|/d\mathbf{p} \quad (39)$$

From the generalized derivation we can see that these three quantities can be calculated from the same convolution, and, in fact, with the same program as the original calculation but substituting the three transformations given in (40)-(42):

$$T^{-1}\{\exp[i\varphi_c(\mathbf{s}, \mathbf{p})]\} \quad (40)$$

$$T^{-1}\{|F_o(\mathbf{s})| \exp[i\varphi_c(\mathbf{s}, \mathbf{p})]\} \quad (41)$$

$$T^{-1}\{|F_c(\mathbf{s}, \mathbf{p})| \exp[i\varphi_c(\mathbf{s}, \mathbf{p})]\} \quad (42)$$

Therefore with three FFT's we can calculate the required gradient of the correlation coefficient.

This particular function has not been implemented in TNT. To do so would only require the creation of the code to calculate the means, the coefficients for the transformations, and a program which would combine the means with the results of the convolutions to produce the final gradient. To perform refinement a program would have to be written to calculate r' for any given model. None of these programming tasks is difficult.

B Evaluation of the gradients of the terms necessary for the implementation of stereochemical restraints

By Dale E. Tronrud and Lynn F. Ten Eyck

B.1 Introduction

In this Appendix we present the equations for the gradients of the stereochemical terms. Most of these are derived by straightforward algebra and no details will be given. Because of the assumptions used in the case of planarity restraints these equations are derived explicitly.

The stereochemistry terms are of the following form [*cf.* equation (2) in the main text]:

$$M(\mathbf{p}) = \sum_b W(b)[Q_o(b) - Q_c(b, \mathbf{p})]^2 \quad (43)$$

The gradients are of the form

$$\partial M(\mathbf{p})/\partial \mathbf{x}_i = -2 \sum_b W(b)[Q_o(b) - Q_c(b, \mathbf{p})] \partial Q_c(b, \mathbf{p})/\partial \mathbf{x}_i. \quad (44)$$

Only the portion of (44) unique to each type of restraint (*i.e.* the derivative of the calculated quantity with respect to the parameters of the model) will be listed in each of the following sections. In each section the coordinates are assumed to be expressed in an orthogonal system.

B.2 Bond lengths

For bond lengths, $Q_c(\mathbf{s})$ is the distance between two atoms. Call these two atoms i and j .

$$\begin{aligned} \partial Q_c(\mathbf{x})/\partial x_i &= [1/Q_c(\mathbf{x})](x_i - x_j) \\ \partial Q_c(\mathbf{x})/\partial x_j &= [-1/Q_c(\mathbf{x})](x_i - x_j) \\ \partial Q_c(\mathbf{x})/\partial y_i &= [1/Q_c(\mathbf{x})](y_i - y_j) \\ \partial Q_c(\mathbf{x})/\partial y_j &= [-1/Q_c(\mathbf{x})](y_i - y_j) \\ \partial Q_c(\mathbf{x})/\partial z_i &= [1/Q_c(\mathbf{x})](z_i - z_j) \\ \partial Q_c(\mathbf{x})/\partial z_j &= [-1/Q_c(\mathbf{x})](z_i - z_j) \end{aligned} \quad (45)$$

B.3 Bond angles

For bond angles, $Q_c(\mathbf{x})$ is the angle defined by three atoms, labeled i , j , k , with j being the central atom.

Let \mathbf{a} be the vector from atom j to atom i , \mathbf{b} the vector from atom k to atom j and \mathbf{c} the vector from atom i to atom k . Then

$$Q_c(\mathbf{x}) = \cos^{-1}[a^2 + b^2 - c^2/2ab]. \quad (46)$$

[Note: In Fortran the argument of the function ACOS is in radians. If $Q_c(\mathbf{s})$ is to be in degrees a conversion factor must be included.]

$$\frac{\partial Q_c(\mathbf{x})}{\partial \mathbf{x}_i} = \frac{-1}{2ab|\sin Q_c(\mathbf{x})|} \left\{ \left[1 - \frac{a}{b} \cos Q_c(\mathbf{x}) \right] \frac{\partial a^2}{\partial \mathbf{x}_i} \right\} \quad (47)$$

$$+ \left[1 - \frac{a}{b} \cos Q_c(\mathbf{x}) \right] \left\{ \frac{\partial b^2}{\partial \mathbf{x}_i} - \frac{\partial c^2}{\partial \mathbf{x}_i} \right\}, \quad (48)$$

$$\begin{aligned} \partial a^2 / \partial i_w &= 2(i_w - j_w) & \partial b^2 / \partial i_w &= 0 \\ \partial c^2 / \partial i_w &= -2(k_w - i_w) & \partial a^2 / \partial j_w &= -2(i_w - j_w) \\ \partial b^2 / \partial j_w &= 2(j_w - k_w) & \partial c^2 / \partial j_w &= 0 \\ \partial a^2 / \partial k_w &= 0 & \partial b^2 / \partial k_w &= -2(j_w - k_w) \\ \partial c^2 / \partial k_w &= 2(k_w - i_w) \end{aligned} \quad (49)$$

where:

$$\begin{aligned} i_1 &= x_i & j_1 &= x_j & k_1 &= x_k \\ i_2 &= y_i & j_2 &= y_j & k_2 &= y_k \\ i_3 &= z_i & j_3 &= z_j & k_3 &= z_k \end{aligned} \quad (50)$$

B.4 Torsion angles

For torsion angles, $Q_c(\mathbf{x})$ is the angle defined by the four atoms labeled i, j, k, l . The quantity is the angle between the normal to the plane defined by atoms i, j, k and the normal to the plane defined by atoms j, k, l .

Let \mathbf{a} be the vector from atom j to atom i , \mathbf{s} the vector from atom j to atom k and \mathbf{b} the vector from atom k to atom l . Also let $\mathbf{r} = \mathbf{s} \times \mathbf{a}$ and $\mathbf{q} = \mathbf{s} \times \mathbf{b}$ where \mathbf{r} is the normal to the plane containing atoms i, j, k and \mathbf{q} is the normal to the plane containing atoms j, k, l .

$$Q_c(\mathbf{x}) = \cos^{-1}(\mathbf{r} \cdot \mathbf{q} / r q). \quad (51)$$

[Note: In Fortran the argument of the function ACOS is in radians. If $Q_c(\mathbf{x})$ is to be in degrees a conversion factor must be included.]

Let

$$\mathbf{e} = [-1/rq |\sin Q_c(\mathbf{x})|][\mathbf{q} - \cos Q_c(\mathbf{x})(q/r)\mathbf{r}] \quad (52)$$

and

$$\mathbf{f} = [-1/rq |\sin Q_c(\mathbf{x})|][\mathbf{r} - \cos Q_c(\mathbf{x})(r/q)\mathbf{q}] \quad (53)$$

Then

$$\partial Q_c(\mathbf{s}) / \partial \mathbf{x}_i = \mathbf{e} \times \mathbf{s} \quad (54)$$

$$\partial Q_c(\mathbf{s}) / \partial \mathbf{x}_j = -\partial Q_c(\mathbf{s}) / \partial \mathbf{x}_i + \mathbf{e} \times \mathbf{a} + \mathbf{f} \times \mathbf{b} \quad (55)$$

$$\partial Q_c(\mathbf{s}) / \partial \mathbf{x}_k = -\partial Q_c(\mathbf{s}) / \partial \mathbf{x}_l - \mathbf{e} \times \mathbf{a} - \mathbf{f} \times \mathbf{b} \quad (56)$$

$$\partial Q_c(\mathbf{s}) / \partial \mathbf{x}_l = \mathbf{f} \times \mathbf{s} \quad (57)$$

B.5 Planarity

For planarity, $Q_c(\mathbf{x})$ is defined as the root mean square (r.m.s.) deviation of the atoms from the best-fit plane.

Let N be the number of atoms in the plane and $\bar{\mathbf{x}}$ the center of mass of the atoms. Consider

$$\mathbf{Q} = \begin{pmatrix} \sum_i (x_i - \bar{x})(x_i - \bar{x}) & \sum_i (x_i - \bar{x})(y_i - \bar{y}) & \sum_i (x_i - \bar{x})(z_i - \bar{z}) \\ \sum_i (y_i - \bar{y})(x_i - \bar{x}) & \sum_i (y_i - \bar{y})(y_i - \bar{y}) & \sum_i (y_i - \bar{y})(z_i - \bar{z}) \\ \sum_i (z_i - \bar{z})(x_i - \bar{x}) & \sum_i (z_i - \bar{z})(y_i - \bar{y}) & \sum_i (z_i - \bar{z})(z_i - \bar{z}) \end{pmatrix} \quad (58)$$

\mathbf{Q} is the moments matrix for the atoms of the plane. The eigenvectors of \mathbf{Q} point along the directions of the principal axes of rotation of this group of atoms. The eigenvalues of \mathbf{Q} are inversely related to the moments of inertia of rotation about the axis defined by the corresponding eigenvector. The axis of rotation with the largest moment (smallest eigenvector) is defined as the normal to the best plane for these atoms.

Let u be the smallest eigenvalue of \mathbf{Q} , \mathbf{n} the eigenvector of \mathbf{Q} corresponding to u and \mathbf{m} ($= \mathbf{n}/n$) be the normalized eigenvector. Then the r.m.s. deviation of the atoms from planarity is

$$Q_c(\mathbf{x}) = \left\{ \sum_i [\mathbf{m} \cdot (\mathbf{x}_i - \bar{\mathbf{x}})]^2 / N \right\}^{1/2} \quad (59)$$

$$\begin{aligned} \partial Q_c(\mathbf{x}) / \partial x_i = & \quad [1/N Q_c(\mathbf{x})] \left\{ \sum_j [\mathbf{m} \cdot (\mathbf{x}_j - \bar{\mathbf{x}})] \right. \\ & \times [\partial \mathbf{m} / \partial x_j \cdot (\mathbf{x}_j - \bar{\mathbf{x}})] \\ & \left. + [1 - (1/N)] [\mathbf{m} \cdot (\mathbf{x}_j - \bar{\mathbf{x}})] \mathbf{m} \right\} \end{aligned} \quad (60)$$

$$\partial \mathbf{m} / \partial x_i = (1/n)(\partial \mathbf{n} / \partial \mathbf{x}_i) - (\mathbf{n}/n^3) \wedge (\mathbf{n} \cdot \partial \mathbf{n} / \partial \mathbf{x}_i) \quad (61)$$

where \wedge is the outer product. It is defined, when \mathbf{a} and \mathbf{b} are column vectors, as $\mathbf{a} \wedge \mathbf{b} = \mathbf{a} \cdot \mathbf{b}^t$.

The calculation of the derivative of the eigenvector with respect to the position of an atom is difficult because eigenvectors are usually determined algorithmically. There is no general equation which expresses the components of the eigenvector of an matrix as a function of the components of that matrix.

However, if one assumes that the off-diagonal elements of \mathbf{Q} are non-zero one can derive an equation for the eigenvector:

$$\begin{aligned} n_1 &= Q_{31}Q_{32} + Q_{21}(u - Q_{33}) \\ n_2 &= Q_{31}Q_{31} + (u - Q_{33})(u - Q_{11}) \end{aligned} \quad (62)$$

$$n_3 = Q_{31}Q_{21} + Q_{32}(u - Q_{11}) \quad (63)$$

The derivatives of n_1 , n_2 and n_3 are simple to derive in terms of the derivatives of the elements of \mathbf{Q} . Because of the complexity of its derivative we have made the assumption that the eigenvalue remains constant during refinement.

$$\partial\mathbf{Q}/\partial x_i = (1 - 1/N) \begin{pmatrix} 2(x_i - \bar{x}) & (y_i - \bar{y}) & (z_i - \bar{z}) \\ (y_i - \bar{y}) & 0 & 0 \\ (z_i - \bar{z}) & 0 & 0 \end{pmatrix} \quad (64)$$

$$\partial\mathbf{Q}/\partial y_i = (1 - 1/N) \begin{pmatrix} 0 & (x_i - \bar{x}) & 0 \\ (x_i - \bar{x}) & 2(y_i - \bar{y}) & (z_i - \bar{z}) \\ 0 & (z_i - \bar{z}) & 0 \end{pmatrix} \quad (65)$$

$$\partial\mathbf{Q}/\partial z_i = (1 - 1/N) \begin{pmatrix} 0 & 0 & (x_i - \bar{x}) \\ 0 & 0 & (y_i - \bar{y}) \\ (x_i - \bar{x}) & (y_i - \bar{y}) & 2(z_i - \bar{z}) \end{pmatrix} \quad (66)$$

The assumption that the off-diagonal elements of \mathbf{Q} are non-zero makes the gradient calculation sensitive to the orientation of the plane. In the program which performs these calculations the problems which might arise are ignored. It is presumed that if by chance the plane lies in a special orientation the movement resulting from the first cycle of refinement will cause it to be displaced and subsequent refinement will function normally.

References

AGARWAL, R., LIFCHITZ, A. & DODSON, E. (1981). In *Refinement of Protein Structures*, edited by P. A. MACHIN, J. W. CAMPBELL & M. ELDER. Science and Engineering Research Council, Warrington, England: Daresbury Laboratory.

AGARWAL, R. C. (1978). *Acta Cryst.* **A34**, 791–809.

- ALBER, T., GRÜTTER, M. G., GRAY, T. M., WOZNIAK, J. A., WEAVER, L. H., CHEN, B.-L., BAKER, E. N. & MATTHEWS, B. W. (1986). In *Protein Structure, Folding and Design*, Vol. 39 of *UCLA Symposia on Molecular and Cellular Biology, New Series*, pp. 307–318. New York: Alan R. Liss, Inc.
- ANDERSON, W. F., OHLENDORF, D. H., TAKEDA, Y. & MATTHEWS, B. W. (1981). *Nature (London)*, **290**, 754–758.
- BOWEN, H. J. M., DONOHUE, J., JENKIN, D. G., KENNARD, O., WHEATLEY, P. J. & WHIFFEN, D. H. (1958). In *Tables of Interatomic Distances and Configuration in Molecules and Ions*, edited by A. D. MITCHELL & L. C. CROSS. London: The Chemical Society.
- DAURAT-LARROQUE, S. T., BREW, K. & FENNA, R. E. (1986). *J. Biol. Chem.* **261**, 3607–3615.
- DIAMOND, R. (1971). *Acta Cryst.* **A27**, 436–452.
- DODSON, E. J., ISAACS, N. W. & ROLLETT, J. S. (1976). *Acta Cryst.* **A32**, 311–315.
- FENNA, R. E. & MATTHEWS, B. W. (1975). *Nature (London)*, **258**, 573–577.
- FLETCHER, R. & REEVES, C. (1964). *Computer Journal*, **7**, 81–84.
- FREER, S. T., ALDEN, R. A., CARTER, C. W. JR & KRAUT, J. (1975). *J. Biol. Chem.* **250**, 46–54.
- GRÜTTER, M. G., WEAVER, L. H., GRAY, T. & MATTHEWS, B. W. (1983). In *Bacteriophage T4*, edited by C. K. MATTHEWS, K. KUTTER, G. MOSIG & P. BERGET, pp. 356–360. Washington DC: American Society for Microbiology.
- HENDRICKSON, W. A. & KONNERT, J. H. (1980). In *Computing in Crystallography*, edited by R. DIAMOND, S. RAMASESHAN & K. VENKATESAN, chapter 13, pp. 13.01–13.26. Bangalore: Indian Academy of Sciences.
- HERMANS, J. JR & MCQUEEN, J. E. JR (1974). *Acta Cryst.* **A30**, 730–739.

- HOLMES, M. A. & MATTHEWS, B. W. (1982). *J. Mol. Biol.* **160**, 623–639.
- HOLMES, M. A., TRONRUD, D. E. & MATTHEWS, B. W. (1983). *Biochemistry*, **22**, 236–240.
- ISAACS, N. (1982). In *Computational Crystallography*, edited by D. SAYRE, pp. 398–408. Oxford: Clarendon.
- JACK, A. & LEVITT, M. (1978). *Acta Cryst.* **A34**, 931–935.
- JONES, T. A. & LILJAS, L. (1984). *Acta Cryst.* **A40**, 50–57.
- KONNERT, J. H. & HENDRICKSON, W. A. (1980). *Acta Cryst.* **A36**, 344–350.
- MATTHEWS, B. W., FENNA, R. E., BOLOGNESI, M. C., SCHMID, M. F. & OLSON, J. M. (1979). *J. Mol. Biol.* **131**, 259–285.
- SUSSMAN, J. L., HOLBROOK, S. R., CHURCH, J. M. & KIM, S. (1977). *Acta Cryst.* **A33**, 800–804.
- TEN EYCK, L. F. (1977). *Acta Cryst.* **A33**, 486–492.
- TEN EYCK, L. F., WEAVER, L. H. & MATTHEWS, B. W. (1976). *Acta Cryst.* **A32**, 349–350.
- TRONRUD, D. E., HOLDEN, H. M. & MATTHEWS, B. W. (1987). *Science*, **235**, 571–574.
- TRONRUD, D. E., MONZINGO, A. F. & MATTHEWS, B. W. (1986). *Eur. J. Biochem.* **157**, 261–268.
- TRONRUD, D. E., SCHMID, M. F. & MATTHEWS, B. W. (1986). *J. Mol. Biol.* **188**, 443–454.
- VIJAYAN, M. (1976). In *CRC Handbook of Biochemistry and Molecular Biology, Proteins, Vol III*, 3rd ed., edited by G. D. FASMAN, pp. 742–759. Cleveland: CRC Press.
- WARME, P. K., GÖ, W. & SCHERAGA, H. A. (1972). *J. Comput. Phys.* **9**, 303–317.

- WATENPAUGH, K. D., SIEKER, L. C., HERRIOTT, J. R. & JENSEN, L. H. (1973). *Acta Cryst.* **B29**, 943–956.
- WLODAWER, A. & HENDRICKSON, W. A. (1982). *Acta Cryst.* **A38**, 239–247.